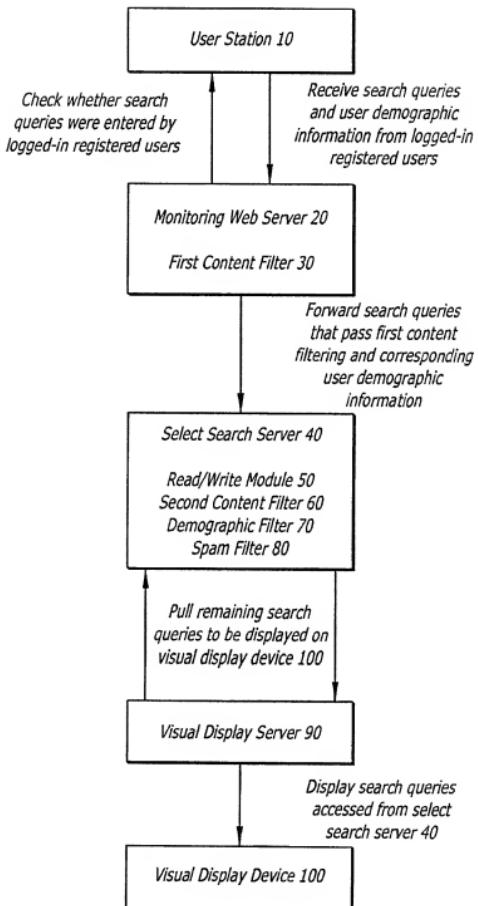


1/18

FIG. 1



2/18

FIG. 2

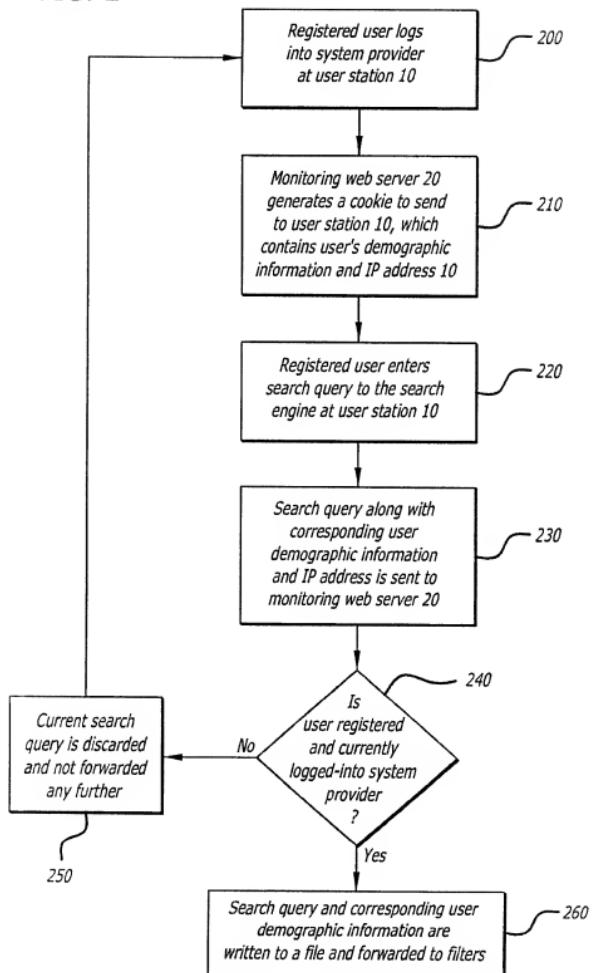


FIG. 3

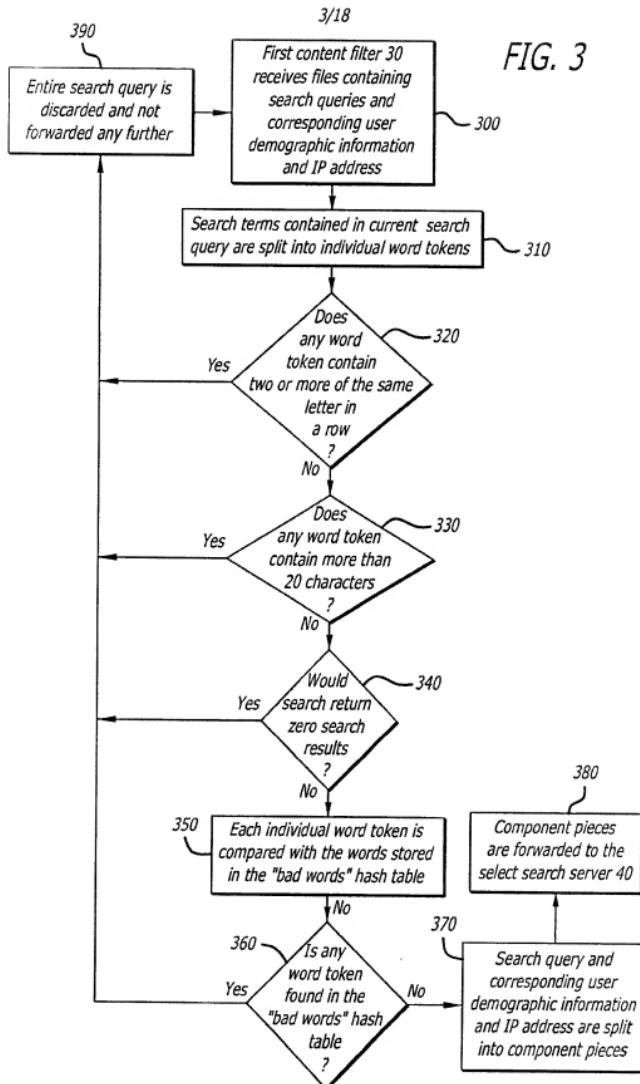


FIG. 4

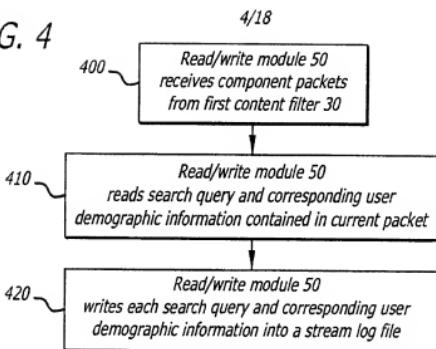
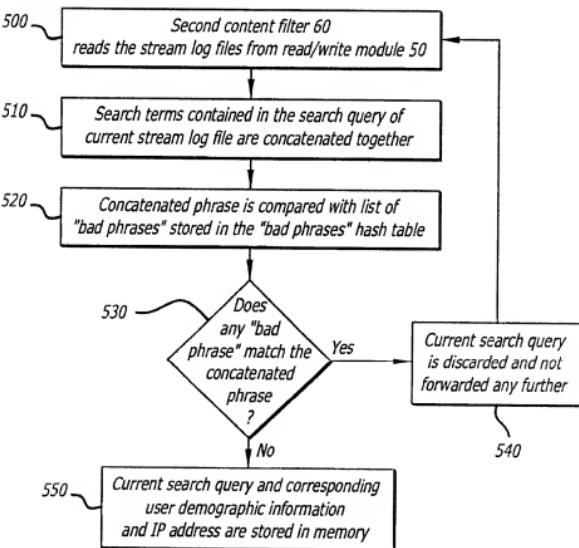
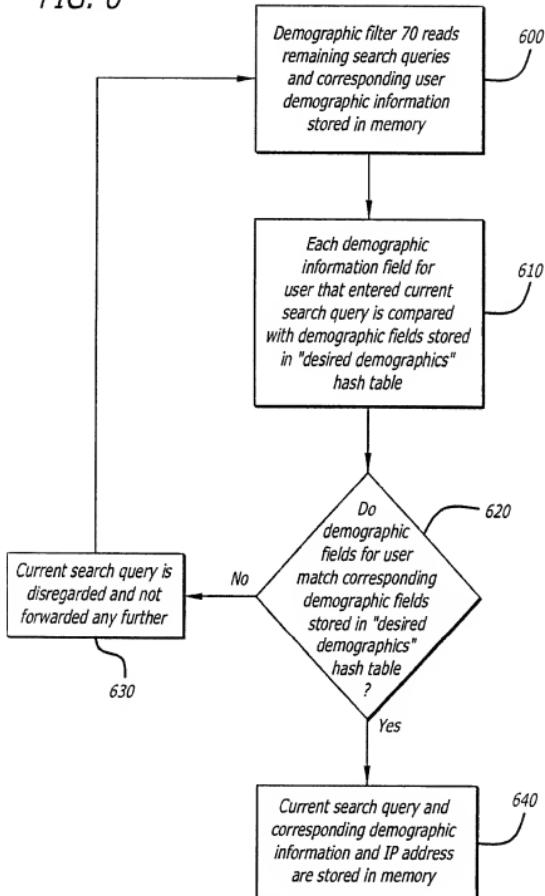


FIG. 5



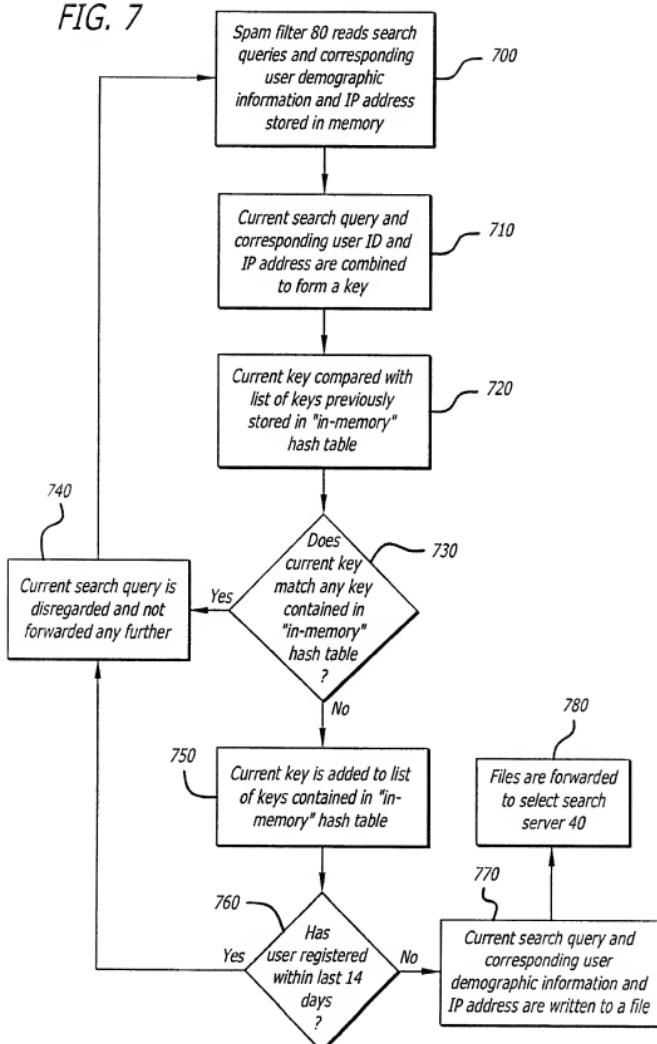
5/18

FIG. 6



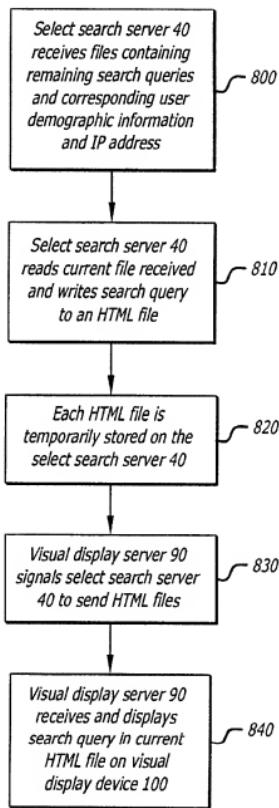
6/18

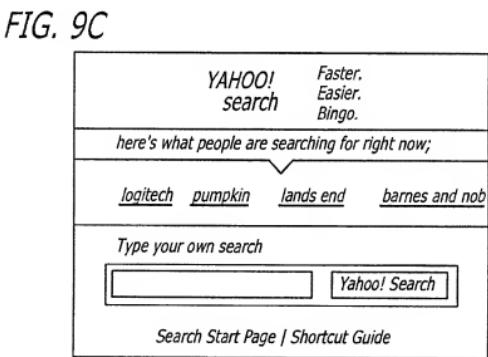
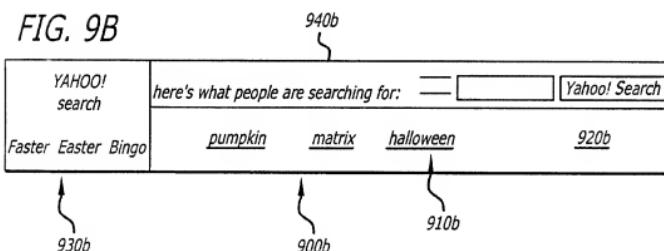
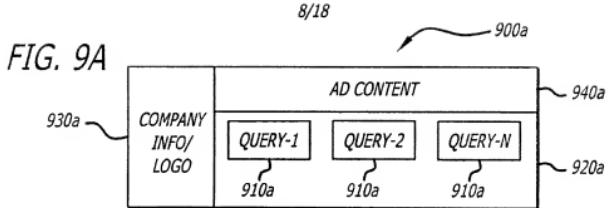
FIG. 7



7/18

FIG. 8





9/18

```

// shim will look for the following variables which will be supplied by the ad unit:
// nqIn = number of terms to get from the CGI - required!
// ageIn = age range (all:0-150/default, 1:0-18, 2:18-25, 3:25-35, 4:35-50, 5:50-150)
// genIn = gender (m-male, f-female, a-all) - not required
// zipIn = zip code - not required
// radIn = radius around zip-code - not required
BLOCK 1:
var extraParams = "";
if ( (nqIn < 1) or (nqIn == undefined) ) nqIn = 20;
if ( ageIn != undefined ) { extraParams += "&age=" + ageIn; }
if ( genIn != undefined ) { extraParams += "&gen=" + genIn; }
if ( zipIn != undefined ) { extraParams += "&zip=" + zipIn; }
if ( radIn != undefined ) { extraParams += "&rad=" + radIn; }

BLOCK 2:
1: baseURL = "http://select.search.server.com/terms?nq=" + nqIn + "&r=x";
2: extraParams += "&rnd=" + Math.random();
3: baseURL += extraParams;

```

BLOCK 3:

```

buzz = new XML();
buzz.onLoad = parseResults;
buzz.ignoreWhite = true;
var listing = [];
buzz.load(baseURL);

```

BLOCK 4:

```
var dataState = "loading";
```

BLOCK 5:

```

function parseResults(result) {
    if (result) {
        var items = buzz.firstChild;
        for (i=1; i < items.length; i++) {
            listing[i-1] = new Object();
            listing[i-1]["keyword"] = items[i].childNodes[0];
        }
        dataState = "available";
    }
    else
    {
        dataState = "unavailable";
    }
}

```

FIG. 10

10/18

```

// This SWF looks for two variables
// delta = step size to take; dy=20 yields 2.0 pixels per
// frame
// offset = distance between keywords; can be negative to
// tighten up spacing between keyword blocks
// If they are not found, default values are set in frame
// 15 of this level.
// ttw = "time to wait" in seconds - defaults to 5 seconds
// (in this frame--see below)

// shim.swf will look for variables as follows:
// n = number of terms to get from the CGI - defaults to 20 if not set
// a = age range (all:0-150/default, 1:0-18, 2:18-25, 3:25-35, 4:35-50, 5:50-150)
// gen = gender (m-male, f-female, a-all) - not required
// zip = zip code - not required
// rad = radius around zip-code - not required
//
// EXAMPLE:
// To get a scroll of 20 keywords from users in ZIP code
// 94089, load this scroller as follows:
//
ticker.loadMovie("http://path_to_scroller_SWF/vscroll_300x300.swf?n=20&zc=9408
9"

```

BLOCK 1:

```
pShim.loadMovie("http://select.search.server.com/shim.swf?nqIn="+ nq
+"&ageIn="+a+"&genIn="+gen+"&zipIn="+zip+"&radIn="+rad)
```

BLOCK 2:

```
var scrollStatus = "loading"
```

BLOCK 3a:

```
var startTime = getTimer()
```

BLOCK 3b:

```
if ( ttw == undefined ){ ttw = 5; }
```

FIG. 11

11/18

BLOCK 1:

```
if ( ttw*1000 < (startTime - getTimer()) )  
  
{  
    scrollStatus = "unavailable";  
    goToAndStop(15);  
}  
else if ( pShim.dataState == "loading" )
```

BLOCK 2:

```
{  
    gotoAndPlay(2);  
    scrollStatus = "loading"  
}  
else
```

BLOCK 3:

```
{  
    scrollStatus = pShim.dataState;  
    gotoAndStop(15);  
}
```

FIG. 12

12/18

```
// delta = step size to take; dy=20 yields 2.0 pixels per frame
// offset = distance between keywords; can be negative to
// tighten up spacing between keyword blocks
BLOCK 1:
if( delta == undefined ){ delta = 20; }
BLOCK 2:
if( offset == undefined ){ offset = 0; }
BLOCK 3:
offset = 1.0*offset;// coerce from string to number, just in case
BLOCK 4:
initMove=move=delta/10

BLOCK 5:
isMoving=true

BLOCK 6:
function hmove(mc){
  if(!isMoving) {
    move=0
  } else (move=initMove  )
}

BLOCK 6a:
mc._x -= move

BLOCK 6b:
if(mc._x < -mc._width){
  mc._x+=2*xPos;    }
  mc._x= Math.floor(mc._x)  }

BLOCK 7:
stop();

BLOCK 8a:
hoverColor="FF0000"
BLOCK 8b:
regularColor="0000FF"
```

FIG. 13a

13/18

BLOCK 9:

searchURL = "http://search.server.com/search?p=";

BLOCK 10:

```
if ( scrollStatus == "available" )  
{  
    var localListing = [];  
    localListing = pShim.listing;  
    formatResults(localListing);  
}
```

BLOCK 11:

```
function formatResults(data) {  
    xPos=0  
    for (i=0; i<data.length; i++) {  
        buzzMC1.attachMovie("item", "b"+i, i);  
        buzzMC2.attachMovie("item", "b"+i, i);  
        var mc1 = buzzMC1["b"+i];  
        var mc2 = buzzMC2["b"+i];  
        var head = data[i].keyword;  
        var url = searchURL+escape(head);  
        mc1.u = mc2.u=url;  
        mc1.keyword = mc2.keyword = head;  
        mc1.head = mc2.head = <font  
        mc1.head = mc2.head = <font  
        color="#"+regularColor+"><u>"+head+"</u></font>";  
        mc1.txt = mc2.txt= head  
        var txtWidth=pixelWidthArial(head, 10);  
        mc1.buttonMC._width=mc2.buttonMC._width = txtWidth  
        mc1._x = mc2._x=xPos;  
        xPos += txtWidth+offset  
    }  
    buzzmc2._x +=xPos  
}
```

FIG. 13b

14/18

BLOCK 1:

```
on(rollOver){  
    _parent._parent._parent.isMoving=false  
    _parent.head = "<font  
color='"+_parent._parent._parent.hoverColor+"'>"+_parent.txt + "</font>"  
}
```

BLOCK 2:

```
on(rollOut, dragOut){ // Block 2  
    _parent._parent._parent.isMoving=TRUE  
    _parent.head = "<font  
color='"+_parent._parent._parent.regularColor+""><u>"+_parent.txt + "</u></font>"  
}
```

BLOCK 3:

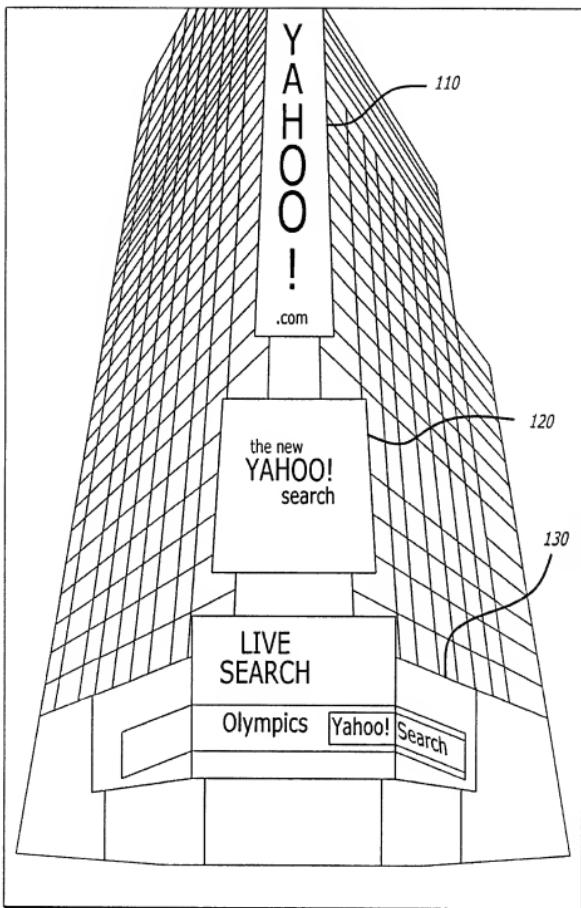
```
on(release){  
    // function doClick(keyword) must be defined in the _root level or nothing happens  
    _root.doClick(_parent.keyword)  
}
```

FIG. 14

15/18

100

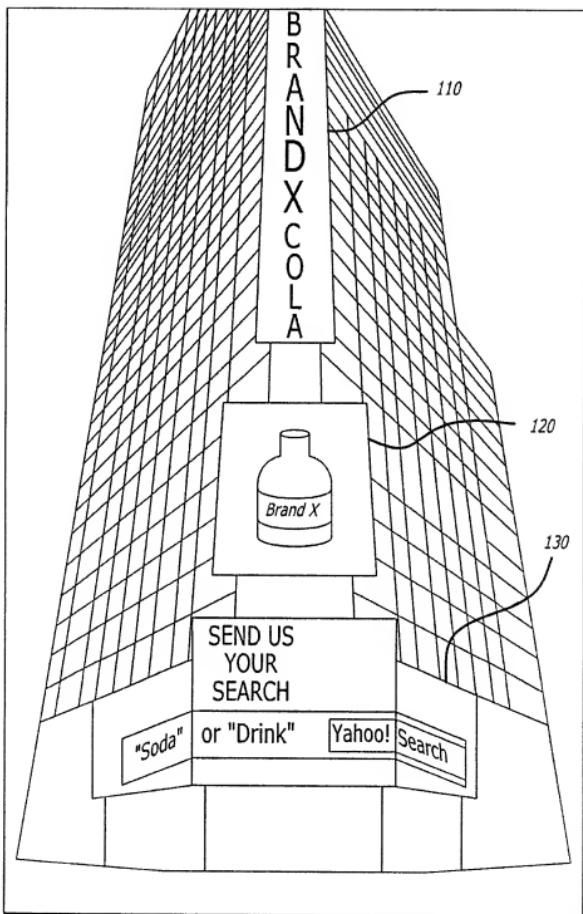
FIG. 15



16/18

100

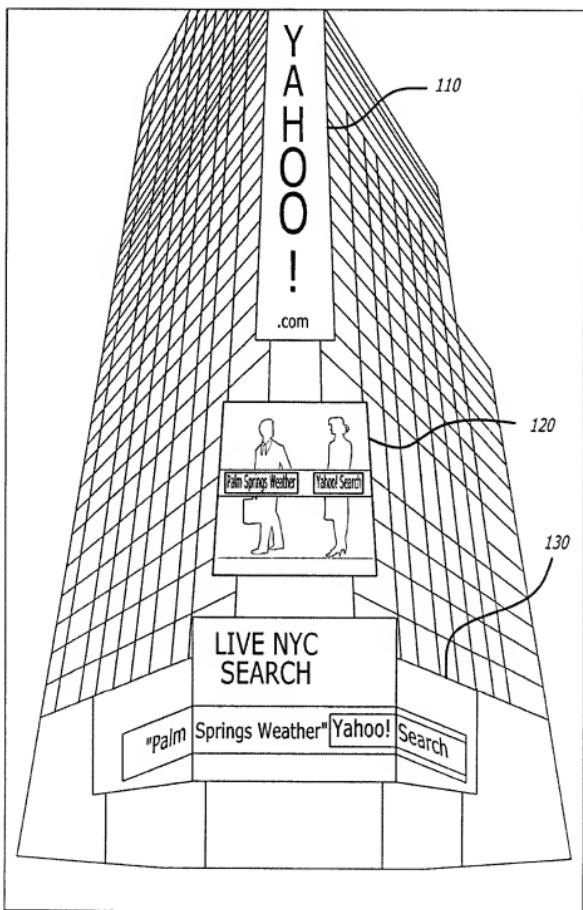
FIG. 16



17/18

100

FIG. 17



18/18

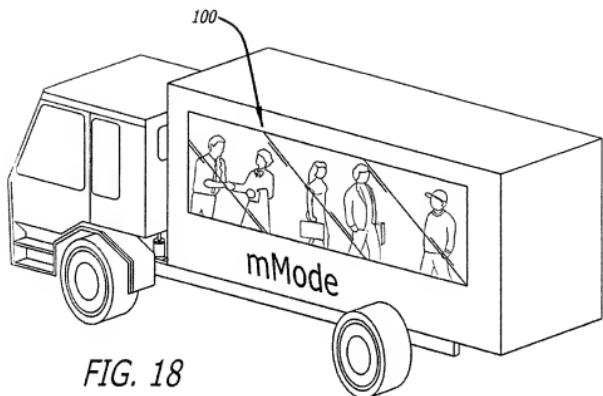


FIG. 18

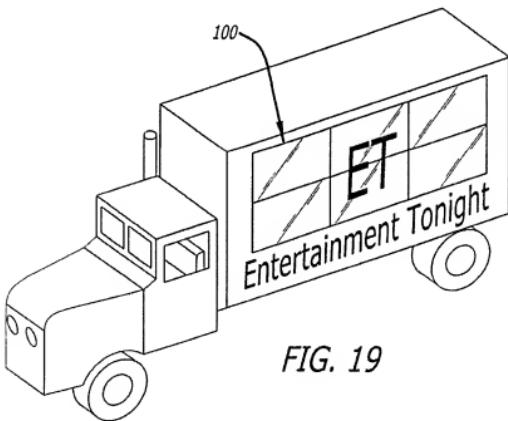


FIG. 19